**РАБОЧАЯ ПРОГРАММА**

|  |  |
| --- | --- |
| Шифр | Наименование дисциплины |
| ***Б3.Б.4*** | ***Программирование*** |

|  |  |
| --- | --- |
| Код направления подготовки | 09.03.01 |
| Направление подготовки | ***Информатика и вычислительная техника*** |
| Наименование ОПОП | ***Системотехника и автоматизация проектирования и управления в строительстве*** |
| Год начала подготовки | ***2015*** |
| Уровень образования | ***Бакалавр*** |
| Форма обучения | ***Очная*** |

**Разработчики:**

|  |  |  |  |
| --- | --- | --- | --- |
| должность | ученая степень, звание | подпись | ФИО |
|  |  |  | ***Садовский Б.С.*** |
|  |  |  |  |

**Рабочая программа рассмотрена и одобрена на заседании кафедры: Информационных систем, технологий и автоматизации в строительстве**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| должность | подпись | | ученая степень и звание, ФИО | | |
| Зав. кафедрой (руководитель подразделения) |  | | **Д.т.н., проф. Гинзбург А.В.** | | |
| год обновления | 2014 | 2015 | | 2016 |  |
| Номер протокола |  |  | |  |  |
| Дата заседания кафедры (структурного подразделения) |  |  | |  |  |

**Рабочая программа утверждена и согласована:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Подразделение / комиссия | Должность | ФИО | подпись | Дата |
| Методическая комиссия |  |  |  |  |
| НТБ |  |  |  |  |
| ЦОСП |  |  |  |  |

1. **Цель освоения дисциплины**

Целью освоения дисциплины «Программирование» является обучить студентов основам техники программирования на алгоритмическом языке высокого уровня. В качестве инструмента последовательно изучается и применяется язык Си. В процессе изучения дисциплины студенты приобретают навыки работы в среде разработчика программ, овладевают основными приемами решения типовых задач, знакомятся с возможностями операционной системы для организации взаимодействия между отдельными программами. Так же целью является изучение теоретических основ и технологии программирования, приобретение практических навыков в разработки ПО, базирующихся на фундаментальных принципах построения программного продукта (ПП) и объектно-ориентированной парадигме программирования (ООП).

1. **Перечень планируемых результатов обучения по дисциплине , соотнесенных с планируемыми результатами освоения образовательной программы**

| Компетенция  по ФГОС | Код компетенции по ФГОС | Основные показатели освоения (показатели достижения результата) | Код  показателя  освоения |
| --- | --- | --- | --- |
| Использует основные законы естественнонаучных дисциплин в профессиональной деятельности, применяет методы математического анализа и моделирования, теоретического и экспериментального исследования | ОК-10 | **Знает** основные математические методы получения, хранения, обработки, передачи и использования информации | З1 |
| **Умеет**применять математический аппарат для решения поставленных задач | У1 |
| **Имеет навыки**навыками использования современного математического аппарата при решении прикладных задач | Н1 |
| Имеет навыки работы с компьютером как средством управления информацией | ОК-12 | **Знает** архитектуру персональных компьютеров и структуру программного обеспечения ПК | З2 |
| **Умеет** составлять программы на языках высокого уровня, обрабатывающие входные данные, взятые из различных электронных носителей, имеющих различные форматы. | У2 |
| **Имеет навыки**владения базовыми математическими знаниями и информационными технологиями | Н2 |
| Способен работать с информацией в глобальных компьютерных сетях | ОК-13 | **Знает** средства поиска информации в Интернет, в различных источниках | З3 |
| **Умеет**оценивать и отбирать наиболее важную информацию, максимально полезную для решения поставленных задач | У3 |
| **Имеет навыки**получения/передачи файлов через Интернет. | Н3 |
| Осваивать методики использования программных средств для решения практических задач | ПК-2 | **Знает** современные технические и программные средства взаимодействия с ЭВМ | З4 |
| **Умеет**составлять программы для решения прикладных инженерных и информационно-поисковых систем (задач). | У4 |
| **Имеет навыки**написания программ на языке высокого уровня | Н4 |
| Разрабатывать интерфейсы «человек – электронно-вычислительная машина» | ПК-3 | **Знает** современные технические и программные средства взаимодействия с ЭВМ | З5 |
| **Умеет**составлять программы с интерфейсом консольного приложения, составлять программы с графическим интерфейсом пользователя | У5 |
| **Имеет навыки**выбора элементной базы для построения различных архитектур вычислительных средств | Н5 |
| Разрабатывать модели компонентов информационных систем, включая модели баз данных | ПК-4 | **Знает** основы системного программирования, основы объектно-ориентированного подхода к программированию | З6 |
| **Умеет**использовать прикладные системы программирования | У6 |
| **Имеет навыки**работы с языками процедурного и объектно-ориентированного программирования | Н6 |
| Разрабатывать компоненты программных комплексов и баз данных, использовать современные инструментальные средства и технологии программирования | ПК-5 | **Знает** технологию разработки алгоритмов и программ, методы отладки и решения задач на ЭВМ в различных режимах | З7 |
| **Умеет**ставить задачу и разрабатывать алгоритм ее решения | У7 |
| **Имеет навыки**разработки и отладки программ не менее, чем на одном из алгоритмических процедурных языков высокого уровня | Н7 |
| Обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности | ПК-6 | **Знает** методы структурного разбиения программы на части, описания алгоритмов в объектно-ориентированной парадигме, отладки и тестирования программного обеспечения (ПО) | З8 |
| **Умеет** моделировать различные объекты с помощью классов; работать с пользовательскими типами данных; использовать рациональные способы проектирования программного продукта | У8 |
| **Имеет навыки** самостоятельного решения конструкторских задач, их алгоритмы и программы реализации с использованием возможностей современной вычислительной техники | Н8 |

1. **Указание места дисциплины в структуре образовательной программы**

Дисциплина «Программирование» относится к профессиональному циклу базовой части учебного плана подготовки бакалавров по направлению 230100 – «Информатика и вычислительная техника», профилям подготовки «Автоматизированные системы обработки информации и управления в строительстве (АСОИУ)», «Системы автоматизации проектирования (САПР) в строительстве» и является обязательной к изучению.

*Необходимая* *подготовка:*

Изучение этой дисциплины не требует от студента знаний, выходящих за пределы программы средней школы, но предъявляет высокие требования к его логической культуре.

*К началу изучения дисциплины студент должен*

*знать:*

- современные технические и программные средства взаимодействия с ЭВМ,

- технологию разработки алгоритмов и программ,

- методы отладки и решения задач на ЭВМ в различных режимах,

- методы и средства компьютерной графики и геометрического моделирования,

- фундаментальные принципы обработки машинной и внемашинной информации,

- базовые принципы кодирования информации,

- процедурный подход, язык высокого уровня Си,

- основные способы хранения данных в ОП,

- математические алгоритмы решения задач, понятия матриц и правил их вычисления;

*уметь:*

- ставить задачу и разрабатывать алгоритм ее решения,

- работать с современными системами программирования, включая объектно-ориентированные,

- решать экономико-математические задачи с использованием ЭВМ,

- кодировать алгоритмы на языке высокого уровня Си,

- использовать математическую библиотеку Си,

- использовать математические модели;

*владеть:*

- языками процедурного и объектно-ориентированного программирования,

- навыками разработки и отладки программ не менее, чем на одном из алгоритмических процедурных языков программирования высокого уровня,

- навыками работы с системой исчисления и внутреннего представления данных в ЭВМ,

- навыками работы с системой ввода/вывода в Си,

- навыками использования библиотечных функций в Си.

*Данные навыки формируются в процессе изучения предшествующих дисциплин:*

«Информатика»,

«Математика».

*Навыки, полученные при изучении дисциплины «Программирование», будут в дальнейшем использованы при изучении дисциплин:*

* «Технология программирования»
* «Геометрическое моделирование»
* «Вычислительная математика»;
* «Кроссплатформенные системы»;
* «Системы искусственного интеллекта»;
* «Базы данных»;
* «Сети и телекоммуникации».

1. **Объем дисциплины в зачетных единицах с указанием количества академических или астрономических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся**

Общая трудоемкость дисциплины составляет 16 зачетных единиц 576 акад.часов.

*(1 зачетная единица соответствует 36 академическим часам)*

*Структура дисциплины:*

Форма обучения - очная

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| №  п/п | Наименование раздела  дисциплины | Семестр | Неделя семестра | Виды учебной работы, включая самостоятельную работу обучающихся и трудоемкость (в часах) | | | | | | Формы текущего контроля  успеваемости  *(по неделям*  *семестра)*  Форма  промежуточной аттестации  *(по семестрам)* |
| Контактная работа с обучающимися | | | | | Самостоятельная работа |
| Лекции | Практико-ориентированные занятия | | | КСР |
| Лабораторный практикум | Практические занятия | Групповые консультации по КП/КР |
| 1. | Общее устройство и приемы работы на ЭВМ. Блок-схемы. Алгоритмический язык Си. | 1 | 1 | 1 |  |  |  |  | 1 |  |
| 2. | Типы величин. | 1 | 2 | 1 |  |  |  |  | 2 |  |
| 3. | Операции (15 рангов) | 1 | 3 | 1 |  |  |  |  | 4 |  |
| 4. | Операторы | 1 | 4 | 1 |  |  |  |  | 4 | Зад№1 |
| 5. | Ввод и вывод. Массивы. | 1 | 5 | 1 |  | 10 |  | 3 | 4 | Зад№2  КР |
| 6. | Работа с файлами | 1 | 6 | 1 |  | 4 |  | 2 | 4 | Зад№3 |
| 7. | Применение пользовательских функций | 1 | 7 | 1 |  | 4 |  | 2 | 4 | Зад№4 |
| 8. | Методы решения нелинейных уравнений | 1 | 8 | 1 |  | 3 |  | 2 | 4 | Зад№5 РГР |
| 9. | Методы поиска экстремума функции | 1 | 9 | 1 |  | 3 |  | 2 | 4 | Зад№5 РГР |
| 10. | Методы численного интегрирования | 1 | 10 | 1 |  | 3 |  | 2 | 4 | Зад№5 РГР |
| 11. | Метод Гаусса для решения СЛАУ | 1 | 11 | 1 |  | 3 |  | 2 | 4 | Зад№5 РГР |
| 12. | Метод Зейделя для решения СЛАУ | 1 | 12 | 1 |  |  |  | 2 | 4 | Зад№5 РГР |
| 13. | Символьные строки | 1 | 13 | 2 |  | 6 |  | 2 | 4 | Зад№6 |
| 14. | Структуры и объединения. | 1 | 15 | 1 |  |  |  | 2 | 4 |  |
| 15. | Графические возможности текстового режима. Программирование меню | 1 | 16-17 | 2 |  |  |  | 2 | 4 |  |
| 16. | Аргументы функции main. Классы памяти | 1 | 18 | 1 |  |  |  | 6 | 8 |  |
|  | **Итого за 1 семестр** | **1** | **18** | **18** |  | **36** |  | **27** | **63** | **Экзамен** |
| 17. | Рекурсия | 2 | 1-2 | 2 |  | 2 |  |  | 2 |  |
| 18. | Организация памяти. Стеки. | 2 | 3-4 | 2 |  | 2 |  |  | 2 |  |
| 19. | Организация памяти. Очереди | 2 | 5-6 | 2 |  | 2 |  |  | 2 |  |
| 20. | Организация памяти. Списки. | 2 | 7-8 | 2 |  | 8 |  |  | 2 |  |
| 21. | Организация памяти. Деревья. | 2 | 9-10 | 2 |  | 2 |  |  | 2 |  |
| 22. | Разбор и выполнение курсовой работы. | 2 | 11-14 | 4 |  | 36 |  | 10 | 26 | Курсовая работа |
| 23. | Графический режим Turbo C. | 2 | 15-16 | 2 |  | 12 |  | 8 | 10 | Зад№7 |
|  | **Итого за 2 семестр** | **2** | **16** | **16** |  | **64** |  | **18** | **46** | **дифференцированный зачет** |
| 24. | История ТП и программирования. Основные понятия и определения. Парадигмы программирования. | 3 | 1-2 | 2 |  |  |  |  | 4 |  |
| 25. | Изучение программирования на языке Си++ | 3 | 3-4 | 2 |  | 16 |  |  | 20 |  |
| 26. | Анализ проблемы (задачи) и синтез спецификаций к ПО | 3 | 5-6 | 2 |  | 16 |  | 9 | 18 | Домашняя работа. |
| 27. | Проектирование и разработка ПО | 3 | 7-8 | 2 |  |  |  |  |  |  |
| 28. | Тестирование и отладка ПО | 3 | 9-10 | 2 |  |  |  |  |  |  |
| 29. | Сопровождение ПО | 3 | 11-12 | 2 |  |  |  |  |  |  |
| 30. | Коллективная разработка. BUG-трекинг. Управление проектом. | 3 | 13-14 | 2 |  |  |  |  |  |  |
| 31. | UML и ГОСТ (ЕСПД). | 3 | 15-16 | 2 |  | 16 |  | 9 | 10 | Контрольная работа. |
| 32. | Форматирование, документация и другие средства стилистической слаженности кода. | 3 | 17-18 | 2 |  |  |  |  |  |  |
|  | **Итого за 3 семестр:** | **3** | **18** | **18** |  | **48** |  | **18** | **52** | **дифференцированный зачет** |
| 33. | Особенности работы с памятью в Си++ (кеш паять). Эффективность производительность. Использование STL. | 4 | 1-6 | 6 |  | 8 |  |  |  |  |
| 34. | Создание программ на Си++ с графическим интерфейсом пользователя | 4 | 7-12 | 6 |  | 32 |  |  | 24 |  |
| 35. | Межпроцессное взаимодействие. Потоки. Взаимодействие программ с ОС. | 4 | 13-16 | 4 |  | 14 |  | 18 | 40 | приём курсовой работы |
|  | **Итого за 4 семестр:** | **4** | **16** | **16** |  | **54** |  | **18** | **64** | **Экзамен** |
|  | **Итого:** | **1, 2, 3,4** | **68** | **68** |  | **202** |  | **81** | **225** | **2 дифференцированных зачета, 2 экзамена** |

1. **Содержание дисциплины , структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий**
   1. *Содержание лекционных занятий*

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Наименование раздела дисциплины | Тема и содержание занятия | Кол-во акад. часов |
| 1. | Общее устройство и приемы работы на ЭВМ. Блок-схемы. Алгоритмический язык Си. | Общее устройство и приемы работы на ЭВМ. Блок-схемы. Обзор современных языков программирования. Алгоритмический язык Си. Алфавит. Зарезервированные слова. | 1 |
| 2. | Типы величин. | Типы величин. Константы. Идентификаторы. Описания. | 1 |
| 3. | Операции (15 рангов) | Операции (15 рангов) в языке Си | 1 |
| 4. | Операторы | Операторы языка Си | 1 |
| 5. | Ввод и вывод. Массивы. | Ввод и вывод. Форматы. Массивы. Выделение памяти. Ввод из файла. Вывод векторов и матриц на экран и в файл. | 1 |
| 6. | Работа с файлами | Работа с файлами. Режимы открытия. Текстовые и двоичные файлы. | 1 |
| 7. | Применение пользовательских функций | Применение пользовательских функций в языке Си. Прототипы функций. Передача значений с использованием адресов и указателей. | 1 |
| 8. | Методы решения нелинейных уравнений | Методы решения нелинейных уравнений: половинного деления, Ньютона, хорд, касательных, хорд и касательных, простой итерации | 1 |
| 9. | Методы поиска экстремума функции | Методы поиска экстремума функции: половинного деления и золотого сечения | 1 |
| 10. | Методы численного интегрирования | Методы численного интегрирования: прямоугольников, трапеций, Симпсона (в т.ч. с экономией вычислений) | 1 |
| 11. | Метод Гаусса для решения СЛАУ | Метод Гаусса для решения СЛАУ (с выбором главного элемента) | 1 |
| 12. | Метод Зейделя для решения СЛАУ | Метод Зейделя для решения СЛАУ | 1 |
| 13. | Символьные строки | Символьные строки. Функции <string.h> | 2 |
| 14. | Структуры и объединения. | Структуры и объединения. | 1 |
| 15. | Графические возможности текстового режима. | Графические возможности текстового режима. Программирование меню. | 2 |
| 16. | Аргументы функции main. Классы памяти | Аргументы функции main. Классы памяти. Внешнее описание объектов. | 1 |
| 17. | Рекурсия | Рекурсия. Факториал. Числа Фибоначчи. Ханойские башни. | 2 |
| 18. | Организация памяти. Стеки. | Организация памяти. Стеки. | 2 |
| 19. | Организация памяти. Очереди | Организация памяти. Очереди | 2 |
| 20. | Организация памяти. Списки. | Организация памяти. Списки. | 2 |
| 21. | Организация памяти. Деревья. | Организация памяти. Деревья. | 2 |
| 22. | Разбор курсовой работы. | Разбор выполнения курсовой работы. | 4 |
| 23. | Графический режим Turbo C. | Графический режим Turbo C. | 2 |
| 24. | История ТП и программирования. Основные понятия и определения. Парадигмы программирования. | Основные понятия и определения. Особенности создания ПО. Классификация ПО. Основные понятия и определения. ТП в истории. Как хранятся данные в ЭВМ и системы исчисления. | 2 |
| 25. | Изучение программирования на языке Си++ | Особенности объектно-ориентированных языков программирования. | 2 |
| 26. | Анализ проблемы (задачи) и синтез спецификаций к ПО | Определение требований к программным продуктам. Функциональные требования. Эксплуатационные требования. Выбор архитектуры ПО. Структура и формат данных. Модульное программирование. Анализ требований и определение спецификаций при структурном подходе. Словарь терминов. Анализ требований и определение спецификаций при объектно-ориентированном подходе. | 2 |
| 27. | Проектирование и разработка ПО | Проектирование при структурном подходе. CASE-технологии. Функциональная схема. Методология RAD. Проектирование при объектно-ориентированном подходе. Диаграммы кооперации. Экстремальное программирование. Рефакторинг. Программирование и тестирование. Инструментальные средства разработки. Выбор языка. Выбор среды. Модели разработки открытая и проприентарная. | 2 |
| 28. | Тестирование и отладка ПО | Тестирование белого ящика и чёрного ящика. Порядок разработки тестов. Модульное тестирование. Интеграционное тестирование. Системное тестирование. Надёжность ПО. Отладка программ. | 2 |
| 29. | Сопровождение ПО | Виды программной документации. Пояснительная записка. Руководство пользователя. Руководство по API для прикладного программиста. | 2 |
| 30. | Коллективная разработка. BUG-трекинг. Управление проектом. | Организация работ при разработке ПО. Ведение проекта по разработке ПО. Прикладное ПО для организации работ по программированию. | 2 |
| 31. | UML и ГОСТ (ЕСПД). | Использование ГОСТ при составлении документации и пояснительной записки. Введение в UML. Основы структурного моделирования. Основы моделирования поведения. Основы моделирования архитектуры. | 2 |
| 32. | Форматирование, документация и другие средства стилистической слаженности кода. | Привила и договорённости в оформлении программного кода. Документирование кода. Декомпозиция. Присвоение имён. Использование языковых средств. Форматирование. | 2 |
| 33. | Особенности работы с памятью в Си++ (кеш паять). Эффективность производительность. Использование STL. | Эффективность и оптимизация программ. Эффективное управление памятью. Проблемы с низкоуровневыми операциями с памятью. Использование кеш памяти и пула объектов. Профилирование программ. Оптимизация использования памяти. | 6 |
| 34. | Создание программ на Си++ с графическим интерфейсом пользователя | Основные модули в разработке ПО. Проблемы безопастности при разработке программ (функции strcpy, strcat). | 6 |
| 35. | Межпроцессное взаимодействие. Потоки. Взаимодействие программ с ОС. | Основные понятия при работе с несколькими потоками и общение процессов между собой. Способы общения Программ с ОС. | 4 |

* 1. *Лабораторный практикум*

Не предусмотрено планом.

* 1. *Перечень практических занятий*

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Наименование раздела дисциплины | Тема и содержание занятия | Кол-во акад. часов |
| 1 | Блок-схемы | Блок схемы. Разбор задач на ветвящиеся алгоритмы. | 10 |
| 2 | Массивы | Разбор задач на массивы. | 4 |
| 3 | Функции в языке Си | Применение пользовательских функций при разработке программ.  Решение задач линейной алгебры. | 4 |
| 4 | Численные методы | Составление программ для численных методов. | 3 |
| 5 | Символьные строки | Решение задач на символьные строки. | 3 |
| 6 | Структуры | Примеры использования структур | 3 |
| 7 | Объединения | Битовые операции и сдвиги. Применение масок. | 3 |
| 8 | Графика. | Графические возможности текстового режима.  Программирование меню | 6 |
| 9 | Рекурсия | Рекурсия | 2 |
| 10 | Стеки | Организация памяти. Стеки. | 2 |
| 11 | Очереди | Организация памяти. Очереди | 2 |
| 12 | Списки | Организация памяти. Списки. | 8 |
| 13 | Деревья | Организация памяти. Деревья. | 2 |
| 14 | Курсовая работа | Разбор и выполнение курсовой работы. | 36 |
| 15 | Графический режим Turbo C. | Графический режим Turbo C. | 12 |
| 16 | Основы создание интерфейса графического приложения | Создать проект в Qt Creator. В проекте с помощью дизайнера создать интерфейс диалогового окна, главного окна, контекстного меню. Написать код позволяющий работать с этими эементами. Реализовать технологию «drag&drop» в вашем проекте. | 14 |
| 17 | Графика 2D | Создать проект в Qt Creator. В нём создать элемент «Canvas». На нём изобразить построение двух мерной фигуры круг. Изобразить лесенку. Элемент круг должен двигаться по лестнице сначало поднимаясь потом спускаясь с неё. | 12 |
| 18 | Графика 3D | Создать проект в Qt Creator. Изобразить трёхмерный элемент шар. Размножить его по кругу 12 раз, для образования циферблата часов. Сделать, что бы каждый шар изменял свой цвет и форму на куб отображая таким образом который сейчас час. | 12 |
| 19 | Базы данных | Создать проект в Qt Creator. Создать Несколько записей в базе данных SQLite. Создать в программе подключение к этой базе. Создать интерфейс пользователя для подключения и отображения содержимого базы. Добавить возможность добавлять и удалять записи в БД. | 14 |
| 20 | Работа с сетью | Создать проект в Qt Creator. Создать клиентское приложение. Создать серверное приложение. В клиенте должна быть возможность вводить текст. Клиент должен подключаться к серверу и обмениваться с ним текстовыми сообщениями. А сервер в свою очередь передает сообщение другому клиенту. Таким образом сообщение проходит маршрут клиент-сервер-клиент. | 14 |
| 21 | Работа с XML | Создать проект в Qt Creator. Написать файл в формате XML. В приложении включить драйвер XML парсера. Написать парсер для чтения XML файла. Сделать возможность редактирования XML файла. | 12 |
| 22 | Многопоточность | Используя проект по работе с базой данных, добавить возможность сортировки записей в таблице в отдельном потоке, с отображением хода сортировки на экране. | 12 |
| 23 | Встроенные системы | Создать проект в Qt Creator по работе с мобильными устройствами. Перенести приложение по обмену сообщениями (клиентская часть) с компьютера на коммуникатор. Клиенты должны через сеть wifi обмениваться сообщениями через сервер. | 12 |

* 1. *Самостоятельная работа*

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Наименование раздела дисциплины | Содержание работы | Кол-во акад. часов |
| 1. | Общее устройство и приемы работы на ЭВМ. Блок-схемы. Алгоритмический язык Си. | Изучение приемов работы в среде Turbo C. | 2 |
| 2. | Типы величин. | Изучение особенностей представления величин различного типа в различных версиях языка Си. | 2 |
| 3. | Операции (15 рангов) | Приобретение навыков записи арифметических выражений на языке Си. | 4 |
| 4. | Операторы | Изучение возможностей операторов языка Си. | 4 |
| 5. | Ввод и вывод. Массивы. | Разбор и решение задач на массивы. | 4 |
| 6. | Работа с файлами | Изучение возможностей чтение и записи информации в файл в различных режимах. | 2 |
| 7. | Применение пользовательских функций | Изучение возможностей применение пользовательских функций в языке Си. | 6 |
| 8. | Методы решения нелинейных уравнений | Изучение методов решения нелинейных уравнений: половинного деления, Ньютона, хорд, касательных, хорд и касательных, простой итерации | 4 |
| 9. | Методы поиска экстремума функции | Изучение методов поиска экстремума функции: половинного деления и золотого сечения | 4 |
| 10. | Методы численного интегрирования | Изучение методов численного интегрирования: прямоугольников, трапеций, Симпсона (в т.ч. с экономией вычислений) | 2 |
| 11. | Метод Гаусса для решения СЛАУ | Изучение метода Гаусса для решения СЛАУ (с выбором главного элемента) | 2 |
| 12. | Метод Зейделя для решения СЛАУ | Изучение метода Зейделя для решения СЛАУ. | 2 |
| 13. | Символьные строки | Изучение возможностей обработки символьных строк и функций библиотеки <string.h> | 4 |
| 14. | Структуры и объединения. | Изучение правил применения в программах на языке Си структур и объединений. | 4 |
| 15. | Графические возможности текстового режима. | Изучение графических возможностей текстового режима. Программирование меню. | 4 |
| 16. | Аргументы функции main. Классы памяти | Изучение классов памяти. Внешнее описание объектов. | 4 |
| 17. | Рекурсия | Изучение возможностей применения рекурсии в языке Си. | 4 |
| 18. | Организация памяти. Стеки. | Изучение возможностей программирования стека в языке Си. | 2 |
| 19. | Организация памяти. Очереди | Изучение возможностей программирования очереди в языке Си. | 4 |
| 20. | Организация памяти. Списки. | Изучение возможностей программирования односторонних и двусторонних списков в языке Си. | 4 |
| 21. | Организация памяти. Деревья. | Изучение возможностей программирования деревьев в языке Си. | 4 |
| 22. | Разбор курсровой работы. | Выполнения курсовой работы. | 24 |
| 23. | Графический режим Turbo C. | Изучение возможностей программирования в языке графическом режиме Turbo C. | 10 |
| 24. | История ТП и программирования. Основные понятия и определения. Парадигмы программирования. | Системы исчисления двоичные, восьмеричные, десятичные, шестнадцатиричные. Способы хранения данных в ЭВМ. | 19 |
| 25. | Изучение программирования на языке Си++ | Проектирование и разработка ПО. Тестирование и отладка ПО. Сопровождение ПО. Коллективная разработка. BUG-трекинг. Управление проектом. Форматирование, документация и другие средства стилистической слаженности кода. Особенности работы с памятью в Си++ (кеш паять). Эффективность производительность. Использование STL. | 20 |
| 26. | Анализ проблемы (задачи) и синтез спецификаций к ПО | Моделирование в язык UML. Абстракции, классы, связи, диаграммы интерфейсы. Составление спецификации к программе. | 19 |
| 27. | UML и ГОСТ (ЕСПД). | Изучение Единой системы программной документации  Подготовка к контрольной работе  Подготовка к сдаче зачета | 19 |
| 28. | Создание программ на Си++ с графическим интерфейсом пользователя | Изучение основ построения пользовательского интерфейса  Подготовка курсовой работы  Изучение следующих тем: «Основы создание интерфейса графического приложения», «Графика 2D», «Графика 3D», «Базы данных», «Работа с сетью», «Работа с XML».  Подготовка курсовой работы | 20 |
| 29. | Межпроцессное взаимодействие. Потоки. Взаимодействие программ с ОС. | Изучение темы «Многопоточность».  Подготовка к защите курсовой работы.  Изучение лекционного материала, и подготовка к экзамену. | 19 |

1. **Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине**

Более сорока процентов времени, отведенного на изучение дисциплины «Программирование на языке высокого уровня» составляет самостоятельная работа студента. Постоянная самостоятельная работа по курсу является залогом усвоения знаний и успешного прохождения всех этапов аттестации предусмотренных рабочей программой по дисциплине.

Самостоятельную работу по курсу можно разделить по следующим направлениям:

* Изучение текущих и дополнительных теоретических вопросов;
* Совершенствование навыков по решению практических задач;
* Подготовка курсовой работы;
* Подготовка к мероприятиям по текущей и итоговой аттестации.

При самостоятельной работе рекомендуется пользоватьсяучебными материалами, указанными в разделе 8.

1. **Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине** 
   1. *Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Код компетенции  по ФГОС | Этапы формирования компетенций (разделы теоретического обучения) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 |
| ОК-10 | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + |
| ОК-12 | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + |
| ОК-13 | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + |
| ПК-2 | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + |
| ПК-3 | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + |
| ПК-4 | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + |
| ПК-5 | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + |
| ПК-6 | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + | + |

* 1. *Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания*
     1. *Описание показателей и форм оценивания компетенций*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Код компетенции по ФГОС | Показатели освоения  (Код показателя освоения) | Форма оценивания | | | | | Обеспеченность оценивания компетенции |
| Текущий контроль | | Промежуточная аттестация | | |
| Домашнее задание | Контрольная работа | Защита курсовой работы/ проекта | Зачет-/дифференцированный зачет | Экзамен |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| ОК-10 | З1 | + | + | + | + | + | + |
| У1 | + | + | + | + | + | + |
| Н1 | + | + | + | + | + | + |
| ОК-12 | З2 | + | + | + | + | + | + |
| У2 | + | + | + | + | + | + |
| Н2 | + | + | + | + | + | + |
| ОК-13 | З3 | + | + | + | + | + | + |
| У3 | + | + | + | + | + | + |
| Н3 | + | + | + | + | + | + |
| ПК-2 | З4 | + | + | + | + | + | + |
| У4 | + | + | + | + | + | + |
| Н4 | + | + | + | + | + | + |
| ПК-3 | З5 | + | + | + | + | + | + |
| У5 | + | + | + | + | + | + |
| Н5 | + | + | + | + | + | + |
| ПК-4 | З6 | + | + | + | + | + | + |
| У6 | + | + | + | + | + | + |
| Н6 | + | + | + | + | + | + |
| ПК-5 | З7 | + | + | + | + | + | + |
| У7 | + | + | + | + | + | + |
| Н7 | + | + | + | + | + | + |
| ПК-6 | З8 | + | + | + | + | + | + |
| У8 | + | + | + | + | + | + |
| Н8 | + | + | + | + | + | + |

* + 1. *Описание шкалы и критериев оценивания для проведения промежуточной аттестации обучающихся по дисциплине в форме Экзамена/Дифференцированного зачета*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Код показателя оценивания | Оценка | | | |
| «2»  (неудовлетв.) | Пороговый уровень освоения | Углубленный уровень освоения | Продвинутый уровень освоения |
| «3»  (удовлетвор.) | «4»  (хорошо) | «5»  (отлично) |
| З1 | Обучающийся не знает значительной части программного материала в части формирования основных принципов программирования на языке высокого уровня. Допускает существенные ошибки. Не ориентируется в материале. | Обучающийся имеет знания только основного материала в части формирования основных принципов программирования на языке высокого уровня, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала. | Обучающийся твердо знает материал программирования на языке высокого уровня. Грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, уверенно ориентируется в материале. | Обучающийся глубоко и прочно усвоил программный материал в части программирования на языке высокого уровня. Исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал из литературы, правильно обосновывает принятое решение. |
| У1 | Обучающийся неуверенно, с большими затруднениями выполняет практические работы, необходимые практические компетенции не сформированы,  не умеет применить теоретические знания при решении практических задач. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при ответе на практические вопросы экзаменатора | Обучающийся правильно применяет теоретические положения при решении практических вопросов и задач. | Обучающийся свободно справляется с задачами, вопросами и другими видами применения знаний. Умеет анализировать полученные результаты; проявляет самостоятельность при выполнении практических заданий. |
| Н1 | Обучающийся не показывает практических навыков в выполнении практических заданий, требуемых составом компетенций. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике, при ответе на практические вопросы экзаменатора. | Обучающийся имеет прочные знания программирования на языке высокого уровня, уверенно применяет теоретические знания. | Обучающийся не только имеет прочные знания программирования на языке высокого уровня, но свободно оперирует материалом из литературы. |
| З2 | Обучающийся не знает значительной части программного материала в части формирования основных принципов программирования на языке высокого уровня. Допускает существенные ошибки. Не ориентируется в материале. | Обучающийся имеет знания только основного материала в части формирования основных принципов программирования на языке высокого уровня, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала. | Обучающийся твердо знает материал программирования на языке высокого уровня. Грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, уверенно ориентируется в материале. | Обучающийся глубоко и прочно усвоил программный материал в части программирования на языке высокого уровня. Исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал из литературы, правильно обосновывает принятое решение. |
| У2 | Обучающийся неуверенно, с большими затруднениями выполняет практические работы, необходимые практические компетенции не сформированы,  не умеет применить теоретические знания при решении практических задач. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при ответе на практические вопросы экзаменатора | Обучающийся правильно применяет теоретические положения при решении практических вопросов и задач. | Обучающийся свободно справляется с задачами, вопросами и другими видами применения знаний. Умеет анализировать полученные результаты; проявляет самостоятельность при выполнении практических заданий. |
| Н2 | Обучающийся не показывает практических навыков в выполнении практических заданий, требуемых составом компетенций. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике, при ответе на практические вопросы экзаменатора. | Обучающийся имеет прочные знания программирования на языке высокого уровня, уверенно применяет теоретические знания. | Обучающийся не только имеет прочные знания программирования на языке высокого уровня, но свободно оперирует материалом из литературы. |
| З3 | Обучающийся не знает значительной части программного материала в части формирования основных принципов программирования на языке высокого уровня. Допускает существенные ошибки. Не ориентируется в материале. | Обучающийся имеет знания только основного материала в части формирования основных принципов программирования на языке высокого уровня, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала. | Обучающийся твердо знает материал программирования на языке высокого уровня. Грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, уверенно ориентируется в материале. | Обучающийся глубоко и прочно усвоил программный материал в части программирования на языке высокого уровня. Исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал из литературы, правильно обосновывает принятое решение. |
| У3 | Обучающийся неуверенно, с большими затруднениями выполняет практические работы, необходимые практические компетенции не сформированы,  не умеет применить теоретические знания при решении практических задач. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при ответе на практические вопросы экзаменатора | Обучающийся правильно применяет теоретические положения при решении практических вопросов и задач. | Обучающийся свободно справляется с задачами, вопросами и другими видами применения знаний. Умеет анализировать полученные результаты; проявляет самостоятельность при выполнении практических заданий. |
| Н3 | Обучающийся не показывает практических навыков в выполнении практических заданий, требуемых составом компетенций. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике, при ответе на практические вопросы экзаменатора. | Обучающийся имеет прочные знания программирования на языке высокого уровня, уверенно применяет теоретические знания. | Обучающийся не только имеет прочные знания программирования на языке высокого уровня, но свободно оперирует материалом из литературы. |
| З4 | Обучающийся не знает значительной части программного материала в части формирования основных принципов программирования на языке высокого уровня. Допускает существенные ошибки. Не ориентируется в материале. | Обучающийся имеет знания только основного материала в части формирования основных принципов программирования на языке высокого уровня, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала. | Обучающийся твердо знает материал программирования на языке высокого уровня. Грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, уверенно ориентируется в материале. | Обучающийся глубоко и прочно усвоил программный материал в части программирования на языке высокого уровня. Исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал из литературы, правильно обосновывает принятое решение. |
| У4 | Обучающийся неуверенно, с большими затруднениями выполняет практические работы, необходимые практические компетенции не сформированы,  не умеет применить теоретические знания при решении практических задач. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при ответе на практические вопросы экзаменатора | Обучающийся правильно применяет теоретические положения при решении практических вопросов и задач. | Обучающийся свободно справляется с задачами, вопросами и другими видами применения знаний. Умеет анализировать полученные результаты; проявляет самостоятельность при выполнении практических заданий. |
| Н4 | Обучающийся не показывает практических навыков в выполнении практических заданий, требуемых составом компетенций. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике, при ответе на практические вопросы экзаменатора. | Обучающийся имеет прочные знания программирования на языке высокого уровня, уверенно применяет теоретические знания. | Обучающийся не только имеет прочные знания программирования на языке высокого уровня, но свободно оперирует материалом из литературы. |
| З5 | Обучающийся не знает значительной части программного материала в части формирования основных принципов программирования на языке высокого уровня. Допускает существенные ошибки. Не ориентируется в материале. | Обучающийся имеет знания только основного материала в части формирования основных принципов программирования на языке высокого уровня, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала. | Обучающийся твердо знает материал программирования на языке высокого уровня. Грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, уверенно ориентируется в материале. | Обучающийся глубоко и прочно усвоил программный материал в части программирования на языке высокого уровня. Исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал из литературы, правильно обосновывает принятое решение. |
| У5 | Обучающийся неуверенно, с большими затруднениями выполняет практические работы, необходимые практические компетенции не сформированы,  не умеет применить теоретические знания при решении практических задач. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при ответе на практические вопросы экзаменатора | Обучающийся правильно применяет теоретические положения при решении практических вопросов и задач. | Обучающийся свободно справляется с задачами, вопросами и другими видами применения знаний. Умеет анализировать полученные результаты; проявляет самостоятельность при выполнении практических заданий. |
| Н5 | Обучающийся не показывает практических навыков в выполнении практических заданий, требуемых составом компетенций. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике, при ответе на практические вопросы экзаменатора. | Обучающийся имеет прочные знания программирования на языке высокого уровня, уверенно применяет теоретические знания. | Обучающийся не только имеет прочные знания программирования на языке высокого уровня, но свободно оперирует материалом из литературы. |
| З6 | Обучающийся не знает значительной части программного материала в части формирования основных принципов программирования на языке высокого уровня. Допускает существенные ошибки. Не ориентируется в материале. | Обучающийся имеет знания только основного материала в части формирования основных принципов программирования на языке высокого уровня, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала. | Обучающийся твердо знает материал программирования на языке высокого уровня. Грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, уверенно ориентируется в материале. | Обучающийся глубоко и прочно усвоил программный материал в части программирования на языке высокого уровня. Исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал из литературы, правильно обосновывает принятое решение. |
| У6 | Обучающийся неуверенно, с большими затруднениями выполняет практические работы, необходимые практические компетенции не сформированы,  не умеет применить теоретические знания при решении практических задач. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при ответе на практические вопросы экзаменатора | Обучающийся правильно применяет теоретические положения при решении практических вопросов и задач. | Обучающийся свободно справляется с задачами, вопросами и другими видами применения знаний. Умеет анализировать полученные результаты; проявляет самостоятельность при выполнении практических заданий. |
| Н6 | Обучающийся не показывает практических навыков в выполнении практических заданий, требуемых составом компетенций. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике, при ответе на практические вопросы экзаменатора. | Обучающийся имеет прочные знания программирования на языке высокого уровня, уверенно применяет теоретические знания. | Обучающийся не только имеет прочные знания программирования на языке высокого уровня, но свободно оперирует материалом из литературы. |
| З7 | Обучающийся не знает значительной части программного материала в части формирования основных принципов программирования на языке высокого уровня. Допускает существенные ошибки. Не ориентируется в материале. | Обучающийся имеет знания только основного материала в части формирования основных принципов программирования на языке высокого уровня, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала. | Обучающийся твердо знает материал программирования на языке высокого уровня. Грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, уверенно ориентируется в материале. | Обучающийся глубоко и прочно усвоил программный материал в части программирования на языке высокого уровня. Исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал из литературы, правильно обосновывает принятое решение. |
| У7 | Обучающийся неуверенно, с большими затруднениями выполняет практические работы, необходимые практические компетенции не сформированы,  не умеет применить теоретические знания при решении практических задач. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при ответе на практические вопросы экзаменатора | Обучающийся правильно применяет теоретические положения при решении практических вопросов и задач. | Обучающийся свободно справляется с задачами, вопросами и другими видами применения знаний. Умеет анализировать полученные результаты; проявляет самостоятельность при выполнении практических заданий. |
| Н7 | Обучающийся не показывает практических навыков в выполнении практических заданий, требуемых составом компетенций. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике, при ответе на практические вопросы экзаменатора. | Обучающийся имеет прочные знания программирования на языке высокого уровня, уверенно применяет теоретические знания. | Обучающийся не только имеет прочные знания программирования на языке высокого уровня, но свободно оперирует материалом из литературы. |
| З8 | Обучающийся не знает значительной части программного материала в части формирования основных принципов программирования на языке высокого уровня. Допускает существенные ошибки. Не ориентируется в материале. | Обучающийся имеет знания только основного материала в части формирования основных принципов программирования на языке высокого уровня, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала. | Обучающийся твердо знает материал программирования на языке высокого уровня. Грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, уверенно ориентируется в материале. | Обучающийся глубоко и прочно усвоил программный материал в части программирования на языке высокого уровня. Исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал из литературы, правильно обосновывает принятое решение. |
| У8 | Обучающийся неуверенно, с большими затруднениями выполняет практические работы, необходимые практические компетенции не сформированы,  не умеет применить теоретические знания при решении практических задач. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при ответе на практические вопросы экзаменатора | Обучающийся правильно применяет теоретические положения при решении практических вопросов и задач. | Обучающийся свободно справляется с задачами, вопросами и другими видами применения знаний. Умеет анализировать полученные результаты; проявляет самостоятельность при выполнении практических заданий. |
| Н8 | Обучающийся не показывает практических навыков в выполнении практических заданий, требуемых составом компетенций. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике, при ответе на практические вопросы экзаменатора. | Обучающийся имеет прочные знания программирования на языке высокого уровня, уверенно применяет теоретические знания. | Обучающийся не только имеет прочные знания программирования на языке высокого уровня, но свободно оперирует материалом из литературы. |

* + 1. *Описание шкалы и критериев оценивания для проведения промежуточной аттестации обучающихся по дисциплине в форме Защиты курсовой работы/проекта*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Код показателя оценивания | Оценка | | | |
| «2»  (неудовлетв.) | Пороговый уровень освоения | Углубленный уровень освоения | Продвинутый уровень освоения |
| «3»  (удовлетвор.) | «4»  (хорошо) | «5»  (отлично) |
| З1 | Обучающийся не знает значительной части программного материала в части программирования на языке высокого уровня. Допускает существенные ошибки. Не может проиллюстрировать полученные знания в процессе выполнения курсовой работы | Обучающийся имеет знания только основного материала в части программирования, но не усвоил его деталей, допускает неточности, что приводит к удовлетворительному уровню выполнения курсовой работы | Обучающийся твердо знает материал в части программирования на языке высокого уровня, что позволяет успешно справляется с задачами, встающими в процессе выполнения курсовой работы. | Обучающийся глубоко и прочно усвоил программный материал в части технологий программирования на языке высокого уровня. Умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний в курсовой работе, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал из литературы, правильно обосновывает принятое решение. |
| У1 | Обучающийся неуверенно, с большими затруднениями выполняет практические работы, необходимые практические компетенции не сформированы,  не умеет применить теоретические знания при решении практических задач, возникающих при выполнении курсовой работы. | Обучающийся имеет знания только основного материала, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при ответе на практические вопросы | Обучающийся правильно применяет теоретические положения при решении практических вопросов и задач, возникающих в процессе выполнения курсовой работы. | Обучающийся свободно справляется с задачами, вопросами и другими видами применения знаний в процессе выполнения курсовой работы. Умеет анализировать полученные результаты; проявляет самостоятельность при выполнении курсовой работы |
| Н1 | Обучающийся не показывает практических навыков в выполнении практических заданий, требуемых составом компетенций. Не может выполнить курсовую работу. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при выполнении курсовой работы | Обучающийся имеет прочные навыки проектной работы в области программирования на языке высокого уровня, для чего уверенно применяет теоретические знания в процессе выполнения курсовой работы. | Обучающийся не только имеет прочные навыки работы в области программирования, но свободно оперирует объемом необходимых знаний в собственной курсовой работе |
| З2 | Обучающийся не знает значительной части программного материала в части программирования на языке высокого уровня. Допускает существенные ошибки. Не может проиллюстрировать полученные знания в процессе выполнения курсовой работы | Обучающийся имеет знания только основного материала в части программирования, но не усвоил его деталей, допускает неточности, что приводит к удовлетворительному уровню выполнения курсовой работы | Обучающийся твердо знает материал в части программирования на языке высокого уровня, что позволяет успешно справляется с задачами, встающими в процессе выполнения курсовой работы. | Обучающийся глубоко и прочно усвоил программный материал в части технологий программирования на языке высокого уровня. Умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний в курсовой работе, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал из литературы, правильно обосновывает принятое решение. |
| У2 | Обучающийся неуверенно, с большими затруднениями выполняет практические работы, необходимые практические компетенции не сформированы,  не умеет применить теоретические знания при решении практических задач, возникающих при выполнении курсовой работы. | Обучающийся имеет знания только основного материала, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при ответе на практические вопросы | Обучающийся правильно применяет теоретические положения при решении практических вопросов и задач, возникающих в процессе выполнения курсовой работы. | Обучающийся свободно справляется с задачами, вопросами и другими видами применения знаний в процессе выполнения курсовой работы. Умеет анализировать полученные результаты; проявляет самостоятельность при выполнении курсовой работы |
| Н2 | Обучающийся не показывает практических навыков в выполнении практических заданий, требуемых составом компетенций. Не может выполнить курсовую работу. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при выполнении курсовой работы | Обучающийся имеет прочные навыки проектной работы в области программирования на языке высокого уровня, для чего уверенно применяет теоретические знания в процессе выполнения курсовой работы. | Обучающийся не только имеет прочные навыки работы в области программирования, но свободно оперирует объемом необходимых знаний в собственной курсовой работе |
| З3 | Обучающийся не знает значительной части программного материала в части программирования на языке высокого уровня. Допускает существенные ошибки. Не может проиллюстрировать полученные знания в процессе выполнения курсовой работы | Обучающийся имеет знания только основного материала в части программирования, но не усвоил его деталей, допускает неточности, что приводит к удовлетворительному уровню выполнения курсовой работы | Обучающийся твердо знает материал в части программирования на языке высокого уровня, что позволяет успешно справляется с задачами, встающими в процессе выполнения курсовой работы. | Обучающийся глубоко и прочно усвоил программный материал в части технологий программирования на языке высокого уровня. Умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний в курсовой работе, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал из литературы, правильно обосновывает принятое решение. |
| У3 | Обучающийся неуверенно, с большими затруднениями выполняет практические работы, необходимые практические компетенции не сформированы,  не умеет применить теоретические знания при решении практических задач, возникающих при выполнении курсовой работы. | Обучающийся имеет знания только основного материала, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при ответе на практические вопросы | Обучающийся правильно применяет теоретические положения при решении практических вопросов и задач, возникающих в процессе выполнения курсовой работы. | Обучающийся свободно справляется с задачами, вопросами и другими видами применения знаний в процессе выполнения курсовой работы. Умеет анализировать полученные результаты; проявляет самостоятельность при выполнении курсовой работы |
| Н3 | Обучающийся не показывает практических навыков в выполнении практических заданий, требуемых составом компетенций. Не может выполнить курсовую работу. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при выполнении курсовой работы | Обучающийся имеет прочные навыки проектной работы в области программирования на языке высокого уровня, для чего уверенно применяет теоретические знания в процессе выполнения курсовой работы. | Обучающийся не только имеет прочные навыки работы в области программирования, но свободно оперирует объемом необходимых знаний в собственной курсовой работе |
| З4 | Обучающийся не знает значительной части программного материала в части программирования на языке высокого уровня. Допускает существенные ошибки. Не может проиллюстрировать полученные знания в процессе выполнения курсовой работы | Обучающийся имеет знания только основного материала в части программирования, но не усвоил его деталей, допускает неточности, что приводит к удовлетворительному уровню выполнения курсовой работы | Обучающийся твердо знает материал в части программирования на языке высокого уровня, что позволяет успешно справляется с задачами, встающими в процессе выполнения курсовой работы. | Обучающийся глубоко и прочно усвоил программный материал в части технологий программирования на языке высокого уровня. Умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний в курсовой работе, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал из литературы, правильно обосновывает принятое решение. |
| У4 | Обучающийся неуверенно, с большими затруднениями выполняет практические работы, необходимые практические компетенции не сформированы,  не умеет применить теоретические знания при решении практических задач, возникающих при выполнении курсовой работы. | Обучающийся имеет знания только основного материала, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при ответе на практические вопросы | Обучающийся правильно применяет теоретические положения при решении практических вопросов и задач, возникающих в процессе выполнения курсовой работы. | Обучающийся свободно справляется с задачами, вопросами и другими видами применения знаний в процессе выполнения курсовой работы. Умеет анализировать полученные результаты; проявляет самостоятельность при выполнении курсовой работы |
| Н4 | Обучающийся не показывает практических навыков в выполнении практических заданий, требуемых составом компетенций. Не может выполнить курсовую работу. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при выполнении курсовой работы | Обучающийся имеет прочные навыки проектной работы в области программирования на языке высокого уровня, для чего уверенно применяет теоретические знания в процессе выполнения курсовой работы. | Обучающийся не только имеет прочные навыки работы в области программирования, но свободно оперирует объемом необходимых знаний в собственной курсовой работе |
| З5 | Обучающийся не знает значительной части программного материала в части программирования на языке высокого уровня. Допускает существенные ошибки. Не может проиллюстрировать полученные знания в процессе выполнения курсовой работы | Обучающийся имеет знания только основного материала в части программирования, но не усвоил его деталей, допускает неточности, что приводит к удовлетворительному уровню выполнения курсовой работы | Обучающийся твердо знает материал в части программирования на языке высокого уровня, что позволяет успешно справляется с задачами, встающими в процессе выполнения курсовой работы. | Обучающийся глубоко и прочно усвоил программный материал в части технологий программирования на языке высокого уровня. Умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний в курсовой работе, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал из литературы, правильно обосновывает принятое решение. |
| У5 | Обучающийся неуверенно, с большими затруднениями выполняет практические работы, необходимые практические компетенции не сформированы,  не умеет применить теоретические знания при решении практических задач, возникающих при выполнении курсовой работы. | Обучающийся имеет знания только основного материала, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при ответе на практические вопросы | Обучающийся правильно применяет теоретические положения при решении практических вопросов и задач, возникающих в процессе выполнения курсовой работы. | Обучающийся свободно справляется с задачами, вопросами и другими видами применения знаний в процессе выполнения курсовой работы. Умеет анализировать полученные результаты; проявляет самостоятельность при выполнении курсовой работы |
| Н5 | Обучающийся не показывает практических навыков в выполнении практических заданий, требуемых составом компетенций. Не может выполнить курсовую работу. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при выполнении курсовой работы | Обучающийся имеет прочные навыки проектной работы в области программирования на языке высокого уровня, для чего уверенно применяет теоретические знания в процессе выполнения курсовой работы. | Обучающийся не только имеет прочные навыки работы в области программирования, но свободно оперирует объемом необходимых знаний в собственной курсовой работе |
| З6 | Обучающийся не знает значительной части программного материала в части программирования на языке высокого уровня. Допускает существенные ошибки. Не может проиллюстрировать полученные знания в процессе выполнения курсовой работы | Обучающийся имеет знания только основного материала в части программирования, но не усвоил его деталей, допускает неточности, что приводит к удовлетворительному уровню выполнения курсовой работы | Обучающийся твердо знает материал в части программирования на языке высокого уровня, что позволяет успешно справляется с задачами, встающими в процессе выполнения курсовой работы. | Обучающийся глубоко и прочно усвоил программный материал в части технологий программирования на языке высокого уровня. Умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний в курсовой работе, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал из литературы, правильно обосновывает принятое решение. |
| У6 | Обучающийся неуверенно, с большими затруднениями выполняет практические работы, необходимые практические компетенции не сформированы,  не умеет применить теоретические знания при решении практических задач, возникающих при выполнении курсовой работы. | Обучающийся имеет знания только основного материала, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при ответе на практические вопросы | Обучающийся правильно применяет теоретические положения при решении практических вопросов и задач, возникающих в процессе выполнения курсовой работы. | Обучающийся свободно справляется с задачами, вопросами и другими видами применения знаний в процессе выполнения курсовой работы. Умеет анализировать полученные результаты; проявляет самостоятельность при выполнении курсовой работы |
| Н6 | Обучающийся не показывает практических навыков в выполнении практических заданий, требуемых составом компетенций. Не может выполнить курсовую работу. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при выполнении курсовой работы | Обучающийся имеет прочные навыки проектной работы в области программирования на языке высокого уровня, для чего уверенно применяет теоретические знания в процессе выполнения курсовой работы. | Обучающийся не только имеет прочные навыки работы в области программирования, но свободно оперирует объемом необходимых знаний в собственной курсовой работе |
| З7 | Обучающийся не знает значительной части программного материала в части программирования на языке высокого уровня. Допускает существенные ошибки. Не может проиллюстрировать полученные знания в процессе выполнения курсовой работы | Обучающийся имеет знания только основного материала в части программирования, но не усвоил его деталей, допускает неточности, что приводит к удовлетворительному уровню выполнения курсовой работы | Обучающийся твердо знает материал в части программирования на языке высокого уровня, что позволяет успешно справляется с задачами, встающими в процессе выполнения курсовой работы. | Обучающийся глубоко и прочно усвоил программный материал в части технологий программирования на языке высокого уровня. Умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний в курсовой работе, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал из литературы, правильно обосновывает принятое решение. |
| У7 | Обучающийся неуверенно, с большими затруднениями выполняет практические работы, необходимые практические компетенции не сформированы,  не умеет применить теоретические знания при решении практических задач, возникающих при выполнении курсовой работы. | Обучающийся имеет знания только основного материала, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при ответе на практические вопросы | Обучающийся правильно применяет теоретические положения при решении практических вопросов и задач, возникающих в процессе выполнения курсовой работы. | Обучающийся свободно справляется с задачами, вопросами и другими видами применения знаний в процессе выполнения курсовой работы. Умеет анализировать полученные результаты; проявляет самостоятельность при выполнении курсовой работы |
| Н7 | Обучающийся не показывает практических навыков в выполнении практических заданий, требуемых составом компетенций. Не может выполнить курсовую работу. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при выполнении курсовой работы | Обучающийся имеет прочные навыки проектной работы в области программирования на языке высокого уровня, для чего уверенно применяет теоретические знания в процессе выполнения курсовой работы. | Обучающийся не только имеет прочные навыки работы в области программирования, но свободно оперирует объемом необходимых знаний в собственной курсовой работе |
| З8 | Обучающийся не знает значительной части программного материала в части программирования на языке высокого уровня. Допускает существенные ошибки. Не может проиллюстрировать полученные знания в процессе выполнения курсовой работы | Обучающийся имеет знания только основного материала в части программирования, но не усвоил его деталей, допускает неточности, что приводит к удовлетворительному уровню выполнения курсовой работы | Обучающийся твердо знает материал в части программирования на языке высокого уровня, что позволяет успешно справляется с задачами, встающими в процессе выполнения курсовой работы. | Обучающийся глубоко и прочно усвоил программный материал в части технологий программирования на языке высокого уровня. Умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний в курсовой работе, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал из литературы, правильно обосновывает принятое решение. |
| У8 | Обучающийся неуверенно, с большими затруднениями выполняет практические работы, необходимые практические компетенции не сформированы,  не умеет применить теоретические знания при решении практических задач, возникающих при выполнении курсовой работы. | Обучающийся имеет знания только основного материала, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при ответе на практические вопросы | Обучающийся правильно применяет теоретические положения при решении практических вопросов и задач, возникающих в процессе выполнения курсовой работы. | Обучающийся свободно справляется с задачами, вопросами и другими видами применения знаний в процессе выполнения курсовой работы. Умеет анализировать полученные результаты; проявляет самостоятельность при выполнении курсовой работы |
| Н8 | Обучающийся не показывает практических навыков в выполнении практических заданий, требуемых составом компетенций. Не может выполнить курсовую работу. | Обучающийся имеет знания только основного материала в части программирования на языке высокого уровня, но не усвоил его деталей, испытывает затруднения в применении теоретических положений на практике при выполнении курсовой работы | Обучающийся имеет прочные навыки проектной работы в области программирования на языке высокого уровня, для чего уверенно применяет теоретические знания в процессе выполнения курсовой работы. | Обучающийся не только имеет прочные навыки работы в области программирования, но свободно оперирует объемом необходимых знаний в собственной курсовой работе |

* + 1. *Описание шкалы и критериев оценивания для проведения промежуточной аттестации обучающихся по дисциплине в форме Зачета*

Не предусмотрено учебным планом.

* 1. *Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций*
     1. *Текущий контроль*

В течении всего обучения ведётся оценка текущей активности обучающихся на основе:

* посещения лекционных и лабораторных занятий;
* изучение работ и отклики на проблемы и предложения сокурсников;
* внятного изложения и восприятия вопросов по теме при консультировании;
* творческого подхода к изучению материала – например, самостоятельный
* поиск источников, конструктивные предложения и др.;
* качества выполнения учебных заданий (с учетом откликов на эти задания);
* соблюдения графика выполнения учебных заданий (лабораторных работ и курсового проекта);
* посещения лекционных и практических занятий;
* изучение работ и отклики на проблемы и предложения сокурсников;
* внятного изложения и восприятия вопросов по теме при консультировании;
* творческого подхода к изучению материала, рекомендованного для самостоятельного изучения;
* качества выполнения практических заданий;
* соблюдения графика выполнения учебных занятий;
* самостоятельную работу по курсу можно разделить по следующим направлениям:
* изучение текущих и дополнительных теоретических вопросов;
* совершенствование навыков по решению практических задач;
* выполнения домашних работ, заданных преподавателем;
* сдачи контрольной работы;
* выполнение и защита курсовой работы;
* подготовка к мероприятиям по текущей и итоговой аттестации.

Для 1-го и 2-го семестров:

Программу дисциплины можно условно разделить на пять частей, в каждой из которых имеются мероприятия текущего контроля:

1. Техника программирования на языке Си. (*Выполнение индивидуальных заданий №1-4,6. Контрольная работа, в рамках которой необходимо написать программу по задаче на массивы*).
2. Применение языка Си для программирования численных методов решения инженерных задач на ЭВМ. (*Оценка знаний: защита индивидуального задания №5*)
3. Работа со структурами. (*Входит составной частью в выполнение курсовой работы*).
4. Организация памяти (стеки, очереди, списки, деревья). (*Входит составной частью в выполнение курсовой работы*).
5. Применение графических средств языка Си. (*Входит составной частью в выполнение курсовой работы + выполнение индивидуального задания №7*).

Примерные вопросы для контрольной работы:

***Задание №1***

Найти максимальное из двух чисел x,y и заменить минимальное нулем:

u = max{x,y}; min{x,y} = 0. Напечатать значения x,y и u .

***Задание №2***

Дан массив C(m), m≤17. Найти произведение всех элементов, предшествующих первой нулевой компоненте, и сумму последующих. Вывести на печать исходный массив и полученные произведение и сумму.

***Задание №3***

Дана матрица A (N, M), которая вводится из файла, и число k, вводимое с клавиатуры. В k-ом столбце найти минимальный элемент. В той строке, где этот элемент находится, найти сумму положительных элементов. Каждый этап решать при помощи подпрограммы, работающей с вектором. Вектор «вырезать» из матрицы.

***Задание №4***

Вычислить скалярное произведение *s. s= (Ap+q, q)*

***Задание №5***

Найти минимум функции  методом золотого сечения

***Задание №6***

В файле хранится список участников соревнования в виде: Фамилия/Год\_рождения (например, Иванов/1985). Кто из них самый молодой?

***Задание №7***

Осуществить такое "движение", при котором окружность движется по другой окружности, оставаясь вне ее.

Для 3-го и 4-го семестров:

При самостоятельной работе рекомендуется пользоваться источниками из списка литературы, приведённого в 7 разделе рабочей программы. Вместе с этим должны быть использованы нормативная документация, справочники, электронные образовательные ресурсы (сетевые образовательные материалы, мультимедийные учебники, энциклопедии и т.п.)

*Контрольная работа.*

Тема: UML и ГОСТ (ЕСПД).

Вариант задания (пример)

1. Разработать схему алгоритма сортировки используя графические обозначения и правила ЕСПД.
2. Разработать схему алгоритма сортировки используя графические обозначения и правила UML.
3. Разработать концептуальную схему БД используя графические обозначения и правила UML сущность-связь.
4. Рассказать про модели программирования.

*Домашнее задание*

Тема: Анализ проблемы (задачи) и синтез спецификаций к разработке ПО.

Подготовить спецификации для следующих задач:

1. Учёт выпускаемой продукции в автоматизированном режиме.
2. Контроль пассажиров, перевозимых в транспортном средстве, в течении дня, недели, месяца.
3. Учёт поставщиков МТР.
4. Расчёт потребности в рабочей силе на строительном объекте.
5. Расчёт прочности строительных конструкций.
6. Составления плана-графика потребности в рабочей силе.

Требуется написать в электронном виде спецификации (ТЗ) по которой можно написать программу. Работа предоставляется к защите.

Вопросы к защите:

1. Где в работе описывается интерфейс функций?
2. Каким образом функции взаимодействуют друг с другом?
3. Что должно быть в разделе «Математическая постановка задачи»?
4. Имеет ли значение используемая программистом парадигма программирования для выработки спецификации?
5. Каким образом влияет?
6. А влияет ли язык программирования на выработку спецификации?
7. Есть ли правила для наименования функций, переменных, классов?
8. Какие ГОСТы можно использовать для написания ТЗ?
9. Как организовать работу нескольких программистов?
10. Какие технические средства можно использовать для составления ТЗ?

2-ой семестр:

*Курсовая работа.*

*Тема -* Разработка информационно-поисковой системы с использованием графических средств языка Си

При выполнении курсовой работы студенты самостоятельно выбирают модель для создания информационно-поисковой системы. В числе обрабатываемых данных необходимо обязательно использовать числовые и строковые типы, структуры. Обязательным является построение двустороннего списка, диаграммы, создания графического интерфейса на основе меню.

Цель курсовой работы состоит в создании информационно-поисковой системы средствами языка Си. В качестве модели может выступать ассортимент магазина, библиотека и т.п. Студенты сами подготавливают базу исходных данных, придумывают (по согласованию с преподавателем) вопросы (пункты меню) и программируют на языке Си графический интерфейс на основе пользовательского меню. В число обязательных вопросов входят создание двустороннего цикла и создание диаграммы (процентное содержание показателей).

Курсовая работа оформляется в виде программы на языке Си, а также в бумажном варианте, в виде компьютерных распечаток текста и иллюстраций на листах формата A4;

*Защита курсовой проекта* проходит в устной форме с проставлением оценки. В ходе защиты студент должен продемонстрировать практические и теоретические знания, полученные в ходе выполнения курсовой работы. В ходе подготовки к защите рекомендуется повторить теоретический материал. Вспомнить и закрепить основные методы, используемые в курсовом проекте.

4-ый семестр:

Курсовая работа.

Изучение материала курса завершается выполнением и защитой курсовой работы на тему «Разработка ПО для подсистемы предприятия строительной отрасли».

Возможно написание работ на темы: учёта, расчёта, хранения, прогнозирования и планирования.

1. Календарный график производства работ на объекте.
2. Расчёт нагрузок ЖБ конструкции.

Ход выполнения работы включает следующие этапы:

- получение задания,

- Программирование модулей

- Тестирование программы

- Оформление пояснительной записки. Курсовая работа оформляется в виде программы на ЭВМ и пояснительной записки в электронном виде в формате PDF.

- Защита.

Защита проводится в устной форме, с демонстрацией, работающей и удовлетворяющей поставленным условиям программы на ЭВМ, предполагается открытая защита в учебной группе. Курсовая работа оценивается следующими оценками: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

*Вопросы к защите курсовой работы:*

1. Объяснить каким образом в программе используется ООП подход.
2. Показать в программе где используется инкапсуляция.
3. Рассказать какие трудности были при проектировании и реализации программы.
4. Где хранятся данные программы после её закрытия?
5. Какая связь существует между типом данных и классом?
6. Для чего в классе используется конструктор и деструктор?
7. Используется в программе процедурный подход и где?
8. Сколько времени понадобилось на разработку ПО и в чём были основные трудности?
9. Каким образом хранятся данные в памяти, во время работы ПО?
10. Какие существуют особенности при написании ПО с графическим интерфейсом?
11. В чём преимущество хранения данных в виде списка или дерева?
12. В чём преимущества использования программы по сравнению с расчётами вручную или с использованием офисных пакетов?
13. Какое программы использовались (кроме IDE) для облегчения написания ПО?
14. Применялись ли какие-либо модели разработки при проектировании и реализации ПО?
    * 1. *Промежуточная аттестация*

Промежуточная аттестация проводится в соответствии с Положением о текущем контроле и промежуточной аттестации.

Промежуточная аттестация, как комплексные мероприятие, включает сдачу зачёта и экзамена по дисциплине. При подготовке к сдаче зачёта (экзамена) рекомендуется пользоваться записями, сделанными на практических занятиях конспектом лекций, а также материалами, полученными в ходе выполнения самостоятельной работы и практических и лабораторных работ.

1-ый семестр:

*Вопросы для экзамена:*

1. Логические операции. Логические выражения.
2. Поразрядные логические операции.
3. Условный оператор. Операция условия.
4. Оператор цикла с предусловием **while**. Прерывание работы цикла.
5. Оператор цикла с постусловием **do-while**. Прерывание работы цикла.
6. Оператор цикла **for**. Прерывание работы цикла.
7. Оператор выбора **switch**.
8. Массивы; их описание, размещение в памяти, инициализация.
9. Строки; их описание, инициализация.
10. Функции для определения длины строки, сравнения и копирования строк.
11. Структуры. Шаблон структуры, имя структуры, указатель на структуру, доступ к элементам структуры.
12. Заказ памяти для одномерного массива.
13. Заказ памяти для матрицы.
14. Заказ памяти для массива структур.
15. Функции; формальные и фактические аргументы. Имя функции как аргумент другой функции.
16. Классы памяти; область действия имен переменных. Блоки.
17. Преобразования данных при помощи функций sprintfи **sscanf**.
18. Аргументы функции main.
19. Файлы; их описание, открытие текстового файла, закрытие файла.
20. Файлы; их описание, открытие двоичного файла, закрытие файла.
21. Чтение из файла, запись в файл (с использованием форматов).
22. Чтение из файла, запись в файл (без формата).
23. Определение текущей позиции в файле; ее установка.
24. Чтение из файла после записи и запись после чтения.
25. Метод деления отрезка пополам для поиска корня функции.
26. Метод касательных.
27. Метод хорд
28. Метод хорд и касательных.
29. Метод итераций для поиска корня функции.
30. Метод прямоугольников (по левым концам).
31. Метод прямоугольников (по правым концам).
32. Метод прямоугольников (метод середин).
33. Метод трапеций.
34. Метод Симпсона.
35. Метод деления отрезка пополам для поиска минимума функции.
36. Метод золотого сечения.

2-ой семестр:

*Вопросы для дифференцированного* *зачета:*

1. Рекурсивное обращение к функции. Вычисление n.
2. Рекурсивное обращение к функции. Числа Фибоначчи.
3. Рекурсивное обращение к функции. Задача о Ханойских башнях.
4. Добавление и удаление числового элемента из стека.
5. Добавление и удаление строки из стека.
6. Создание упорядоченного списка с односторонними ссылками из чисел.
7. Создание упорядоченного списка с двусторонними ссылками из чисел. ­­­
8. Создание упорядоченного списка с односторонними ссылками из строк.
9. Создание упорядоченного списка с двусторонними ссылками из строк.
10. Удаление числа из списка с односторонними ссылками.
11. Удаление строки из списка с односторонними ссылками.
12. Печать двоичного дерева.
13. Формирование двоичного дерева из чисел.
14. Добавление и удаление строки из очереди.

3-ий семестр:

**Список вопросов к зачету**

1. Парадигмы программирования (процедурное, функциональное, логическое и т. д.).
2. Этапы преобразования программы с языка высокого уровня в машинные команды (компиляция, трансляция, интерпретация, связывание, ассемблер).
3. Организация проекта по программированию (файлы \*.h, \*.c), компиляция (gcc), связывание (статическое, динамическое).
4. Понятие потоков ввода/вывода и их использование (stdin, stdout, stderr).
5. Запуск внешних приложений из программы (функция system()).
6. Возвращаемое значение и аргументы функции main() (использование).
7. Функции scanf(), printf().
8. Некоторые функции по работе со строками (strcpy(), strcmp(), strlen(), strcat(), atoi(), sprintf()  ).
9. Файловый ввод/вывод в Си.
10. Объекты cin и cout.
11. Файловый ввод/вывод в Си++.
12. Тип данных string. Основные операции над строками типа string.
13. Тип данных bool. Тип данных void.
14. Понятие пространства имён (namespace) и директивы/объявления using.
15. Препроцессор (#include, #define, #ifdef/#ifndef, #endif).
16. Объявление и инициализация указателей. Опасность при работе с указателем. Указатель на указатель. Указатель на статическую переменную. Указатель на динамическую переменную. Связь массива и указателя.
17. Объявление и инициализация ссылок. Особенности работы ссылок.
18. Динамическое выделение и освобождение памяти (Си и Си++).
19. Операторы членства “.” и “->” используемые при работе с составными типами данных.
20. Общий вид функции. Прототипирование и вызов функции. Передача параметров в функцию. Аргументы функции по умолчанию.
21. Встраиваемые функции. Возвращаемое значение.
22. Спецификаторы, квалификаторы.
23. Автоматическая, статическая и динамическая продолжительность хранения.
24. Внутренние и внешние связывание. Область видимости.
25. Понятие интерфейса.
26. Понятие класса и объекта.
27. Данные и методы.
28. Описание и реализация функций-членов.
29. Конструктор. Объявление и определение конструктора. Использование конструктора. Конструктор по умолчанию. Список инициализаторов.
30. Деструктор. Объявление и использование деструктора.
31. Классы и структуры.
32. Использование констант с объектами.
33. Неявные функции-члены/операции.
34. Указатель this.
35. Дружественные функции.
36. Дружественные классы.
37. Дружественные функции-члены.
38. Предварительное объявление.
39. Спецификаторы доступа: public, private, protected.
40. Понятие инкапсуляции. Процедурный и объектно-ориентированный подходы в программировании.
41. Модели отношений is-a и has-a.
42. Механизм наследование в классах. Понятия базового и производного классов. Передача данных из базового класса в производный.
43. Отношения между базовым и производными классами.
44. Переопределение функций-членов в классах потомках. Спецификатор доступа: protected.
45. Множественное наследование. Проблема множественного наследования.
46. Приватное наследование.
47. Механизм включения.
48. Перегрузка функции.
49. Перегрузка операций.
50. Полиморфное наследование.
51. Виртуальные функции понятия и реализация.
52. Чисто виртуальные функции. Абстрактный базовый класс.
53. Механизм исключений. Блоки try, catch, trow.
54. Шаблоны функций и классов.
55. Библиотека STL (контейнеры, итераторы).
56. Обобщённое программирование.
57. Метод проектирования снизу-вверх. Недостатки указанного метода. Класс решаемых задач.
58. Программирование и разработка ПО. Основная трудность оценки программы.
59. Структурное программирование, и его основные операторы.
60. Метод проектирования сверху вниз. Недостатки указанного метода. Класс решаемых задач.
61. Жизненный цикл ПО.
62. Каскадная модель разработки ПО.
63. Тестирование, отладка и верификация.

Зачёт проводится в устной форме, включает подготовку, ответ студентов на теоретические вопросы и вопросы, рассматриваемые на практических занятиях. К зачету допускаются студенты, успешно защитившие курсовую работу.

4-ый семестр:

**Перечень вопросов к экзамену**

1. Парадигмы программирования (процедурное, функциональное, логическое и т. д.).
2. Этапы преобразования программы с языка высокого уровня в машинные команды (компиляция, трансляция, интерпретация, связывание, ассемблер).

Организация проекта по программированию (файлы \*.h, \*.c), компиляция (gcc), связывание (статическое, динамическое).

1. Понятие потоков ввода/вывода и их использование (stdin, stdout, stderr).
2. Запуск внешних приложений из программы (функция system()).
3. Возвращаемое значение и аргументы функции main() (использование).
4. Функции scanf(), printf().
5. Некоторые функции по работе со строками (strcpy(), strcmp(), strlen(), strcat(), atoi(), sprintf()  ).
6. Файловый ввод/вывод в Си.
7. Объекты cin и cout.
8. Файловый ввод/вывод в Си++.
9. Тип данных string. Основные операции над строками типа string.
10. Тип данных bool. Тип данных void.
11. Понятие пространства имён (namespace) и директивы/объявления using.
12. Препроцессор (#include, #define, #ifdef/#ifndef, #endif).
13. Объявление и инициализация указателей. Опасность при работе с указателем. Указатель на указатель. Указатель на статическую переменную. Указатель на динамическую переменную. Связь массива и указателя.
14. Объявление и инициализация ссылок. Особенности работы ссылок.
15. Динамическое выделение и освобождение памяти (Си и Си++).
16. Операторы членства “.” и “->” используемые при работе с составными типами данных.
17. Общий вид функции. Прототипирование и вызов функции. Передача параметров в функцию. Аргументы функции по умолчанию.
18. Встраиваемые функции. Возвращаемое значение.
19. Спецификаторы, квалификаторы памяти.
20. Автоматическая, статическая и динамическая продолжительность хранения переменных.
21. Внутренние и внешние связывание. Область видимости.
22. Понятие интерфейса.
23. Понятие класса и объекта.
24. Данные и методы.
25. Описание и реализация функций-членов.
26. Конструктор. Объявление и определение конструктора. Использование конструктора. Конструктор по умолчанию. Список инициализаторов.
27. Деструктор. Объявление и использование деструктора.
28. Классы и структуры.
29. Использование констант с объектами.
30. Неявные функции-члены/операции.
31. Указатель this.
32. Дружественные функции.
33. Дружественные классы.
34. Дружественные функции-члены.
35. Предварительное объявление.
36. Спецификаторы доступа: public, private, protected.
37. Понятие инкапсуляции. Процедурный и объектно-ориентированный подходы в программировании.
38. Модели отношений is-a и has-a.
39. Механизм наследование в классах. Понятия базового и производного классов. Передача данных из базового класса в производный.
40. Отношения между базовым и производными классами.
41. Переопределение функций-членов в классах потомках. Спецификатор доступа: protected.
42. Множественное наследование. Проблема множественного наследования.
43. Приватное наследование.
44. Механизм включения.
45. Перегрузка функции.
46. Перегрузка операций.
47. Полиморфное наследование.
48. Виртуальные функции понятия и реализация.
49. Чисто виртуальные функции. Абстрактный базовый класс.
    1. *Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций*

Процедура промежуточной аттестации проходит в соответствии с Положением о текущем контроле и промежуточной аттестации обучающихся.

* Аттестационные испытания проводятся преподавателем, ведущим лекционные занятия по данной дисциплине. Присутствие посторонних лиц в ходе проведения аттестационных испытаний без разрешения ректора или проректора не допускается (за исключением работников университета, выполняющих контролирующие функции в соответствии со своими должностными обязанностями). В случае отсутствия ведущего преподавателя аттестационные испытания проводятся преподавателем, назначенным письменным распоряжением по кафедре (структурному подразделению).
* Инвалиды и лица с ограниченными возможностями здоровья, имеющие нарушения опорно-двигательного аппарата, допускаются на аттестационные испытания в сопровождении ассистентов-сопровождающих.
* Во время аттестационных испытаний обучающиеся могут пользоваться программой учебной дисциплины, а также с разрешения преподавателя справочной и нормативной литературой, калькуляторами.
* Время подготовки ответа при сдаче экзамена в устной форме должно составлять не менее 40 минут (по желанию обучающегося ответ может быть досрочным). Время ответа – не более 15 минут.
* При подготовке к устному экзамену экзаменуемый, как правило, ведет записи в листе устного ответа, который затем (по окончании экзамена) сдается экзаменатору.
* При проведении устного экзамена экзаменационный билет выбирает сам экзаменуемый в случайном порядке.
* Экзаменатору предоставляется право задавать обучающимся дополнительные вопросы в рамках программы дисциплины текущего семестра, а также, помимо теоретических вопросов, давать задачи, которые изучались на практических занятиях.
* Оценка результатов устного аттестационного испытания объявляется обучающимся в день его проведения.
* Оценка по курсовому проекту выставляется на основании результатов защиты на комиссии обучающимся курсового проекта при непосредственном участии преподавателей кафедры (структурного подразделения), руководителя курсового проекта, с возможным присутствием других обучающихся из учебной группы. Одной из форм защиты может быть презентация курсового проекта. Результаты защиты (оценка) вносятся в аттестационную ведомость курсового проекта с указанием темы курсового проекта, а также в зачетную книжку в раздел «Курсовые проекты (работы)».

Процедура защиты курсовой работы (проекта) определена Положением о курсовых работах (проектах).

1. **Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| №  п/п | Наименование дисциплины в соответствии с учебным планом | Автор, название, место издания, издательство, год издания учебной и учебно-методической литературы,  количество страниц | Количество  экземпляров  печатных изданий | Число  обучающихся,  одновременно  изучающих  дисциплину  (модуль) |
| 1 | 2 | 3 | 4 | 5 |
| *Основная литература:* | | | | |
|  |  | НТБ |  |  |
|  | Программирование | Казанский, А. А.   Объектно-ориентированное программирование на Visual Basic 2010 и Visual C# 2010 в среде разработки Microsoft Visual Studio [Текст] : учебное пособие и практикум / А. А. Казанский ; [рец.: В. В. Серов, В. С. Варников] ; Моск. гос. строит. ун-т. - Москва : МГСУ, 2012. - 399 с. | 25 | 90 |
|  | Программирование | Хорев, П. Б.   Объектно-ориентированное программирование [Текст] : учебное пособие для студентов, обучающихся по направлению "Информатика и вычислительная техника" / П. Б. Хорев. - 4-е изд., стериотип. - Москва : Академия, 2012. - 447 с. | 15 | 90 |
|  | Программирование | Казанский, А. А.  Объектно-ориентированое программирование на языке Visual Basic 2008 в среде разработки Microsoft Visual Studio 2008 и NET Framework [Текст] / А. А. Казанский. - Москва : МГСУ.  Ч.1. - 2010.142 с. | 10 | 90 |
|  |  | ЭБС АСВ |  |  |
|  |  |  |  |  |
| *Дополнительная литература:* | | | | |
|  |  | НТБ |  |  |
|  | Программирование | Казанский, А. А.  Объектно-ориентированное программирование на языке Visual Basic 2008 в среде разработки Microsoft Visual Studio 2008 и .NET Framework [Текст] : учеб. пособие и практикум : в 2 ч. / А. А. Казанский ; Моск. гос. строит. ун-т. - М. : МГСУ, 2010.  Ч.2 / [рец.: В. В. Серов, В. С. Варников]. - 2010. - 104 с. | 11 | 90 |
|  | Программирование | Павловская , Т. А.   С/С++. Процедурное и объектно-ориентированное программирование [Текст] : учебник для студентов высших учебных заведений, обучающихся по направлению подготовки дипломированных специалистов "Информатика и вычислительная техника" / Татьяна Павловская . - Москва [и др.] : Питер, 2015. - 495 с. | 25 | 90 |
|  | Программирование | Голицына, О. Л.  Программирование на языках высокого уровня [Текст]: учебное пособие для среднего проф. образования / О. Л. Голицына, И. И. Попов; [рец: И. В. Прохоров [и др.]. - М.: Форум, 2012. - 495 с. | 50 | 90 |

1. **Перечень ресурсов информационно-телекоммуникационной сети «Интернет» (далее – сеть «Интернет), необходимых для освоения дисциплины**

|  |  |
| --- | --- |
| Наименование ресурса сети «Интернет» | Электронный адрес ресурса |
| «Российское образование» - федеральный портал | http://www.edu.ru/index.php |
| Научная электронная библиотека | http://elibrary.ru/defaultx.asp? |
| Электронная библиотечная система IPRbooks | http://www.iprbookshop.ru/ |
| Федеральная университетская компьютерная сеть России | http://www.runnet.ru/ |
| Информационная система "Единое окно доступа к образовательным ресурсам" | http://window.edu.ru/ |
| Научно-технический журнал по строительству и архитектуре «Вестник МГСУ» | http://www.vestnikmgsu.ru/ |
| Научно-техническая библиотека МГСУ | http://www.mgsu.ru/resources/Biblioteka/ |
| раздел «Кафедры» на официальном сайте МГСУ | http://www.mgsu.ru/universityabout/Struktura/Kafedri/ |

1. **Методические указания для обучающихся по освоению дисциплины**

|  |
| --- |
| Организация деятельности обучающегося. |
| 1. Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения, помечать важные мысли, выделять ключевые слова, термины. 2. Ознакомление с терминами, понятиями с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. 3. Определение вопросов, терминов, материала, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удается разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на консультации, на практическом занятии. 4. Уделить внимание следующим понятиям (перечисление понятий) и др. 5. Прослушивание аудио- и видеозаписей по заданной теме. 6. Просмотр рекомендуемой литературы. |
| 1. Знакомство с основной и дополнительной литературой, включая справочные издания, зарубежные источники, конспект основных положений, терминов, сведений, требующихся для запоминания и являющихся основополагающими в этой теме. Составление аннотаций к прочитанным литературным источникам и др. 2. Работа с конспектом лекций, подготовка ответов к контрольным вопросам. 3. Поиск литературы и составление библиографии, использование от 3 до 5 научных работ, изложение мнения авторов и своего суждения по выбранному вопросу, изложение основных аспектов проблемы. 4. Изучение научной, учебной, нормативной и другой литературы. Отбор необходимого материала для написания курсового проекта; формирование выводов и разработка конкретных рекомендаций по решению поставленной цели и задачи; проведение практических исследований по данной теме. Конкретные требования по выполнению и оформлению курсового проекта находятся в методических материалах по дисциплине. 5. Работа с конспектом лекций, подготовка ответов к контрольным вопросам и др. 6. При подготовке к экзамену необходимо ориентироваться на конспекты лекций, рекомендуемую литературу и др. |

1. **Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине , включая перечень программного обеспечения и информационных справочных систем (при необходимости)**
   1. *Перечень информационных технологий, используемых при осуществлении образовательного процесса*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| №  п/п | Наименование раздела дисциплины | Тема | Информационные технологии | Степень обеспеченности (%) |
| 1 | Общее устройство и приемы работы на ЭВМ. Блок-схемы. Алгоритмический язык Си. | Общее устройство и приемы работы на ЭВМ. Блок-схемы. Обзор современных языков программирования. Алгоритмический язык Си. Алфавит. Зарезервированные слова. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 2 | Типы величин. | Типы величин. Константы. Идентификаторы. Описания. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 3 | Операции (15 рангов) | Операции (15 рангов) в языке Си | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 4 | Операторы | Операторы языка Си | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 5 | Ввод и вывод. Массивы. | Ввод и вывод. Форматы. Массивы. Выделение памяти. Ввод из файла. Вывод векторов и матриц на экран и в файл. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 6 | Работа с файлами | Работа с файлами. Режимы открытия. Текстовые и двоичные файлы. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 7 | Применение пользовательских функций | Применение пользовательских функций в языке Си. Прототипы функций. Передача значений с использованием адресов и указателей. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 8 | Методы решения нелинейных уравнений | Методы решения нелинейных уравнений: половинного деления, Ньютона, хорд, касательных, хорд и касательных, простой итерации | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 9 | Методы поиска экстремума функции | Методы поиска экстремума функции: половинного деления и золотого сечения | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 10 | Методы численного интегрирования | Методы численного интегрирования: прямоугольников, трапеций, Симпсона (в т.ч. с экономией вычислений) | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 11 | Метод Гаусса для решения СЛАУ | Метод Гаусса для решения СЛАУ (с выбором главного элемента) | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 12 | Метод Зейделя для решения СЛАУ | Метод Зейделя для решения СЛАУ | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 13 | Символьные строки | Символьные строки. Функции <string.h> | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 14 | Структуры и объединения. | Структуры и объединения. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 15 | Графические возможности текстового режима. | Графические возможности текстового режима. Программирование меню. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 16 | Аргументы функции main. Классы памяти | Аргументы функции main. Классы памяти. Внешнее описание объектов. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 17 | Рекурсия | Рекурсия. Факториал. Числа Фибоначчи. Ханойские башни. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 18 | Организация памяти. Стеки. | Организация памяти. Стеки. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 19 | Организация памяти. Очереди | Организация памяти. Очереди | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 20 | Организация памяти. Списки. | Организация памяти. Списки. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 21 | Организация памяти. Деревья. | Организация памяти. Деревья. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 22 | Разбор курсовой работы. | Разбор выполнения курсовой работы. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 23 | Графический режим Turbo C. | Графический режим Turbo C. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 24 | История ТП и программирования. Основные понятия и определения. Парадигмы программирования. | Основные понятия и определения. Особенности создания ПО. Классификация ПО. Основные понятия и определения. ТП в истории. Как хранятся данные в ЭВМ и системы исчисления. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 25 | Изучение программирования на языке Си++ | Особенности объектно-ориентированных языков программирования. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 26 | Анализ проблемы (задачи) и синтез спецификаций к ПО | Определение требований к программным продуктам. Функциональные требования. Эксплуатационные требования. Выбор архитектуры ПО. Структура и формат данных. Модульное программирование. Анализ требований и определение спецификаций при структурном подходе. Словарь терминов. Анализ требований и определение спецификаций при объектно-ориентированном подходе. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 27 | Проектирование и разработка ПО | Проектирование при структурном подходе. CASE-технологии. Функциональная схема. Методология RAD. Проектирование при объектно-ориентированном подходе. Диаграммы кооперации. Экстремальное программирование. Рефакторинг. Программирование и тестирование. Инструментальные средства разработки. Выбор языка. Выбор среды. Модели разработки открытая и проприентарная. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 28 | Тестирование и отладка ПО | Тестирование белого ящика и чёрного ящика. Порядок разработки тестов. Модульное тестирование. Интеграционное тестирование. Системное тестирование. Надёжность ПО. Отладка программ. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 29 | Сопровождение ПО | Виды программной документации. Пояснительная записка. Руководство пользователя. Руководство по API для прикладного программиста. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 30 | Коллективная разработка. BUG-трекинг. Управление проектом. | Организация работ при разработке ПО. Ведение проекта по разработке ПО. Прикладное ПО для организации работ по программированию. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 31 | UML и ГОСТ (ЕСПД). | Использование ГОСТ при составлении документации и пояснительной записки. Введение в UML. Основы структурного моделирования. Основы моделирования поведения. Основы моделирования архитектуры. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 32 | Форматирование, документация и другие средства стилистической слаженности кода. | Привила и договорённости в оформлении программного кода. Документирование кода. Декомпозиция. Присвоение имён. Использование языковых средств. Форматирование. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 33 | Особенности работы с памятью в Си++ (кеш паять). Эффективность производительность. Использование STL. | Эффективность и оптимизация программ. Эффективное управление памятью. Проблемы с низкоуровневыми операциями с памятью. Использование кеш памяти и пула объектов. Профилирование программ. Оптимизация использования памяти. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 34 | Создание программ на Си++ с графическим интерфейсом пользователя | Основные модули в разработке ПО. Проблемы безопастности при разработке программ (функции strcpy, strcat). | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |
| 35 | Межпроцессное взаимодействие. Потоки. Взаимодействие программ с ОС. | Основные понятия при работе с несколькими потоками и общение процессов между собой. Способы общения Программ с ОС. | Использование слайд-презентаций при проведении лекционных, практических занятий | **100** |

* 1. *Перечень программного обеспечения, используемого при осуществлении образовательного процесса*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование раздела дисциплины | Тема | Наименование программного обеспечения | Тип лицензии |
| 1 | Общее устройство и приемы работы на ЭВМ. Блок-схемы. Алгоритмический язык Си. | Общее устройство и приемы работы на ЭВМ. Блок-схемы. Обзор современных языков программирования. Алгоритмический язык Си. Алфавит. Зарезервированные слова. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 2 | Типы величин. | Типы величин. Константы. Идентификаторы. Описания. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 3 | Операции (15 рангов) | Операции (15 рангов) в языке Си | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 4 | Операторы | Операторы языка Си | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 5 | Ввод и вывод. Массивы. | Ввод и вывод. Форматы. Массивы. Выделение памяти. Ввод из файла. Вывод векторов и матриц на экран и в файл. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 6 | Работа с файлами | Работа с файлами. Режимы открытия. Текстовые и двоичные файлы. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 7 | Применение пользовательских функций | Применение пользовательских функций в языке Си. Прототипы функций. Передача значений с использованием адресов и указателей. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 8 | Методы решения нелинейных уравнений | Методы решения нелинейных уравнений: половинного деления, Ньютона, хорд, касательных, хорд и касательных, простой итерации | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 9 | Методы поиска экстремума функции | Методы поиска экстремума функции: половинного деления и золотого сечения | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 10 | Методы численного интегрирования | Методы численного интегрирования: прямоугольников, трапеций, Симпсона (в т.ч. с экономией вычислений) | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 11 | Метод Гаусса для решения СЛАУ | Метод Гаусса для решения СЛАУ (с выбором главного элемента) | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 12 | Метод Зейделя для решения СЛАУ | Метод Зейделя для решения СЛАУ | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 13 | Символьные строки | Символьные строки. Функции <string.h> | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 14 | Структуры и объединения. | Структуры и объединения. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 15 | Графические возможности текстового режима. | Графические возможности текстового режима. Программирование меню. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 16 | Аргументы функции main. Классы памяти | Аргументы функции main. Классы памяти. Внешнее описание объектов. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 17 | Рекурсия | Рекурсия. Факториал. Числа Фибоначчи. Ханойские башни. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 18 | Организация памяти. Стеки. | Организация памяти. Стеки. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 19 | Организация памяти. Очереди | Организация памяти. Очереди | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 20 | Организация памяти. Списки. | Организация памяти. Списки. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 21 | Организация памяти. Деревья. | Организация памяти. Деревья. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 22 | Разбор курсовой работы. | Разбор выполнения курсовой работы. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 23 | Графический режим Turbo C. | Графический режим Turbo C. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 24 | История ТП и программирования. Основные понятия и определения. Парадигмы программирования. | Основные понятия и определения. Особенности создания ПО. Классификация ПО. Основные понятия и определения. ТП в истории. Как хранятся данные в ЭВМ и системы исчисления. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 25 | Изучение программирования на языке Си++ | Особенности объектно-ориентированных языков программирования. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 26 | Анализ проблемы (задачи) и синтез спецификаций к ПО | Определение требований к программным продуктам. Функциональные требования. Эксплуатационные требования. Выбор архитектуры ПО. Структура и формат данных. Модульное программирование. Анализ требований и определение спецификаций при структурном подходе. Словарь терминов. Анализ требований и определение спецификаций при объектно-ориентированном подходе. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 27 | Проектирование и разработка ПО | Проектирование при структурном подходе. CASE-технологии. Функциональная схема. Методология RAD. Проектирование при объектно-ориентированном подходе. Диаграммы кооперации. Экстремальное программирование. Рефакторинг. Программирование и тестирование. Инструментальные средства разработки. Выбор языка. Выбор среды. Модели разработки открытая и проприентарная. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 28 | Тестирование и отладка ПО | Тестирование белого ящика и чёрного ящика. Порядок разработки тестов. Модульное тестирование. Интеграционное тестирование. Системное тестирование. Надёжность ПО. Отладка программ. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 29 | Сопровождение ПО | Виды программной документации. Пояснительная записка. Руководство пользователя. Руководство по API для прикладного программиста. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 30 | Коллективная разработка. BUG-трекинг. Управление проектом. | Организация работ при разработке ПО. Ведение проекта по разработке ПО. Прикладное ПО для организации работ по программированию. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 31 | UML и ГОСТ (ЕСПД). | Использование ГОСТ при составлении документации и пояснительной записки. Введение в UML. Основы структурного моделирования. Основы моделирования поведения. Основы моделирования архитектуры. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 32 | Форматирование, документация и другие средства стилистической слаженности кода. | Привила и договорённости в оформлении программного кода. Документирование кода. Декомпозиция. Присвоение имён. Использование языковых средств. Форматирование. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 33 | Особенности работы с памятью в Си++ (кеш паять). Эффективность производительность. Использование STL. | Эффективность и оптимизация программ. Эффективное управление памятью. Проблемы с низкоуровневыми операциями с памятью. Использование кеш памяти и пула объектов. Профилирование программ. Оптимизация использования памяти. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 34 | Создание программ на Си++ с графическим интерфейсом пользователя | Основные модули в разработке ПО. Проблемы безопастности при разработке программ (функции strcpy, strcat). | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |
| 35 | Межпроцессное взаимодействие. Потоки. Взаимодействие программ с ОС. | Основные понятия при работе с несколькими потоками и общение процессов между собой. Способы общения Программ с ОС. | Microsoft Windows  Microsoft Visual Studio  Code::Blocks  Qt5 Toolkit 5 | DreamSpark subscription  DreamSpark subscription  Свободное ПО  Свободное ПО |

* 1. *Перечень информационных справочных систем*

Информационно-библиотечные системы

|  |  |
| --- | --- |
| Наименование ИБС | Электронный адрес ресурса |
| Научная электронная библиотека | http://elibrary.ru/defaultx.asp? |
| Электронная библиотечная система IPRbooks | http://www.iprbookshop.ru/ |
| Научно-техническая библиотека МГСУ | http://www.mgsu.ru/resources/Biblioteka/ |

1. **Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине :**

Учебные занятия по дисциплине «Программирование» проводятся в следующих оборудованных учебных кабинетах, оснащенных соответствующим оборудованием и программным обеспечением:

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Вид учебного занятия | Наименование оборудования | № и наименование оборудованных учебных кабинетов, объектов для проведения практических занятий |
| 1 | 2 | 3 | 4 |
| 1 | Лекция | Стационарные / мобильные переносные наборы демонстрационного оборудования; | Аудитория для проведения занятий лекционного типа в соответствии с перечнем аудиторного фонда |
| 2 | Практическое занятие | Мобильные (переносные) наборы демонстрационного оборудования;  Компьютерный класс, оснащенный компьютерами тип №3; | Аудитория для проведения занятий семинарского типа в соответствии с перечнем аудиторного фонда |
| 3 | Практическое занятие | Мобильные (переносные) наборы демонстрационного оборудования;  Автоматизированное рабочее место | Аудитория для проведения занятий семинарского типа в соответствии с перечнем аудиторного фонда |

Программа составлена в соответствии с требованиями Федерального государственного образовательного стандарта высшего профессионального образования с учетом рекомендаций и примерной основной профессиональной образовательной программой высшего профессионального образования по направлению подготовки 09.03.01 «Информатика и вычислительная техника», профиль «Системотехника и автоматизация проектирования и управления в строительстве».